Autonomous Agents, FFR125/FIM760
Lp 3-4, 2007

Teachers:
Krister Wolff and David Sandberg

Robotics assignment 2: Robotic mapping

Time and location:

May 27:th. Location and exact starting time will be annonced on the web page prior to the contest.

Description:

The task of your robot is to explore a given robot arena and generate a metric description (i.e. a map) of the arena. The arena will be in the form of a labyinth with walls of approximately 30 cm height (the walls will be of the same type as in the previous task, mine field navigation). The representation of the map should be on the form "2D occupancy grid", see references [1, 2]. The size of each grid cell should be 30x30 cm.

Apart from the walls there will also be a number of (identical) objects in the arean, which the robot should try to identify (the loccation of) as well. The objects will be in the form of blank metal cylinders with the diameter of 10.5 cm and 27.0 cm height. Points will be given for each correctly identified cylinder in the map, as well as for each correctly classified grid cell (see below).

The map should be coded according to the following scheme: *0=unknown*, *1=free, 2=occupied, 3=object*. That is, each *state* is associated with a number (*0, 1, 2,* or *3*). The robot should navigate around in the arena and use its sensors in order to determine the correct state for each one of the (virtual) grid cells in the 2D occupancy grid map. Initially, all grid cells are classified as *unknown*, i.e. we do not know the state of such a cell. The state *free* means that the robot is free to pass through that cell, *occupied* means that the cell is occuped with an obstacle (i.e. in this case a wall), and *object* means that the cell contains one of the searched metal objects.

The above described coding scheme applies to the map which you should submit to the teachers, who will determine your robots score in the contest. You may of course use another "internal" representation of the map during the actual map generation process. Your map may contain certain intermediate, or probabilistic states, see e.g. [1, 2, 3]. Thus, you are allowed to process the map after it has been generated, e.g. using Matlab or other software, before you submit it. However, no cheating will be accepted! Along with your final map (containing only the above described four distinct states) you should also submit the "raw data" of your map and a description of your internal encoding scheme. Any program (Matlab or other) that you used for processing the raw data and generating the final map should also be included.

1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 0, 0, 0, 2,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 0, 0, 0, 2,
1, 1, 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2,
1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2,
2, 2, 2, 2, 2, 2, 2, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 2,
1, 1, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2,
1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 3, 1, 1, 1, 1, 1, 1, 1, 2,
1, **S**, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2,
1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2,

Fig. 1: Example of a 2D grid map. The encoding scheme is described in the text above. The arena (which is to be covered by the map) is 3 x 6 m, with the surrounding walls *not* included in the map. They will be placed approximately 1/2 grid cell *outside* the area covered by the map. The 2D grid map should contain 10 x 20 = 200 grid cells. The starting position is labeled "**S**" in this map. Thus, that position is known on before hand. Walls and objects contained in the arena will be placed approximately centered on the grid cells, and *not* on the border of two grid cells. The corners in the arena will be N x 90 degrees, where N is an integer.

Execution:

The robot will placed in the grid cell labeled "**S**" with arbitrary heading, which can be chosen by the competing team. On a given command the timekeeping starts. Then, a member of the competing team is allowed to start the robot. Once the robot has started to move, no one is allowed to touch the robot. During a time period of 10 minutes, the robot is allowed to explore the map and build a map of it. Each team will get three trials to map the arena.

The map which the competing team considers to be the best one should then be submitted to the teachers. It is *not* allowed to submit a superposition of the three maps! Some (well defined) processing of the raw data is allowed (see above) in order to generate the final map, but *no* manual modifications of the maps are allowed. The raw data from all three trials *must* be submitted!

Your map will be judged according to the following: One point for each correctly classified cell, and 20 points for each correctly classified *object* in the map will be given. If there are more cells in the map classified as *objects,* than there are objects in the arena, there will be 20 points deduction for *each* faulty object. Further, there will be one point reduction for each cell which is wrongly classified as *free.*

The robot should not run into any obstacles during exploration. If that happens anyway, the robot will be taken out of the arena after the third collision.

Directly after each trial the raw map data should be emailed to the teachers (krister.wolff@chalmers.se, david.sandberg@chalmers.se). Your final map (the one you consider as your robot's best map) should be emailed within 30 minutes after your last trial has finished. Each team will only be given three time slots for their trials. You must be there on time!

The format of the final map you submit for judging should be exactly as shown in the example in Fig.1. That is, comma-separated values, 10 rows and 20 columns. It should be sent via email as an ascii-text file, named "map.group.x.txt", where x is a number between 1 and 9.

Finally, a good overview of the field of robotic mapping is given in [4].

GOOD LUCK!

References

[1] Elfes, A.: Using Occupancy Grids for Mobile Robot Perception and Navigation. *Computer*, **22**(6), 1989.

[2] Moravec, H., and Elfes, A.: High resolution maps from wide angle sonar. In: proceedings of the 1985 IEEE International Conference on Robotics and Automation. Vol.2, pp. 116-121, 1985

[3] Carpin, S. Kenn, H. and Birk, A.: Autonomous Mapping in the Real Robots Rescue League. In proceedings of the RoboCup Symposium, 2003.

[4] Thrun, S.: Robotic mapping: A survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002.